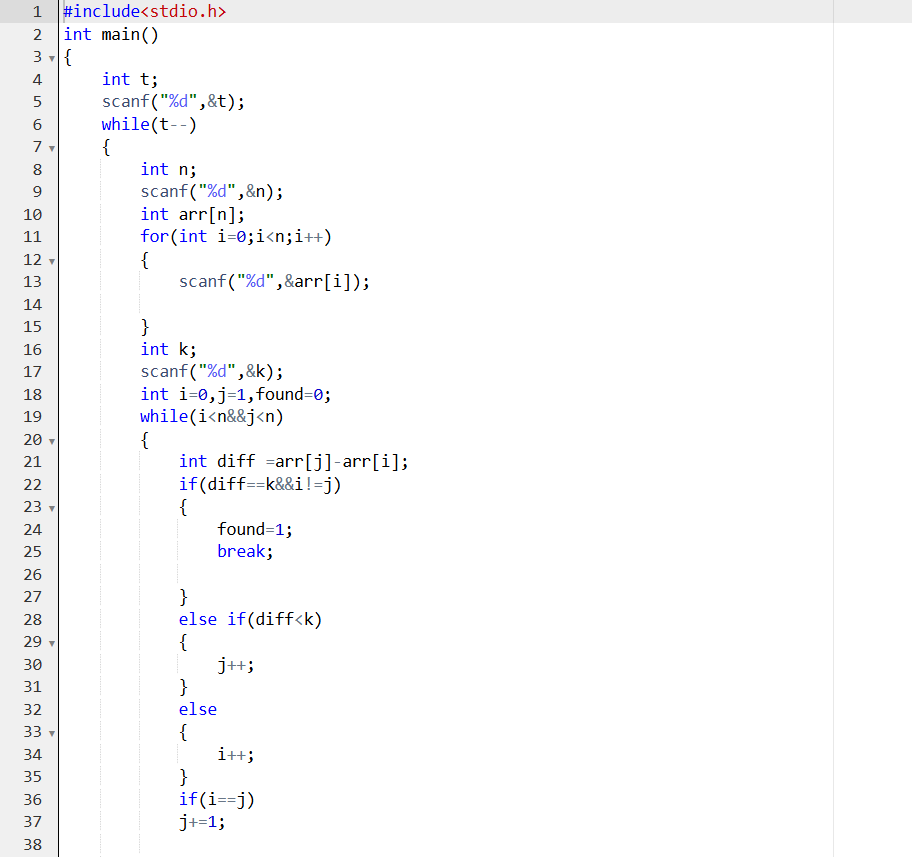
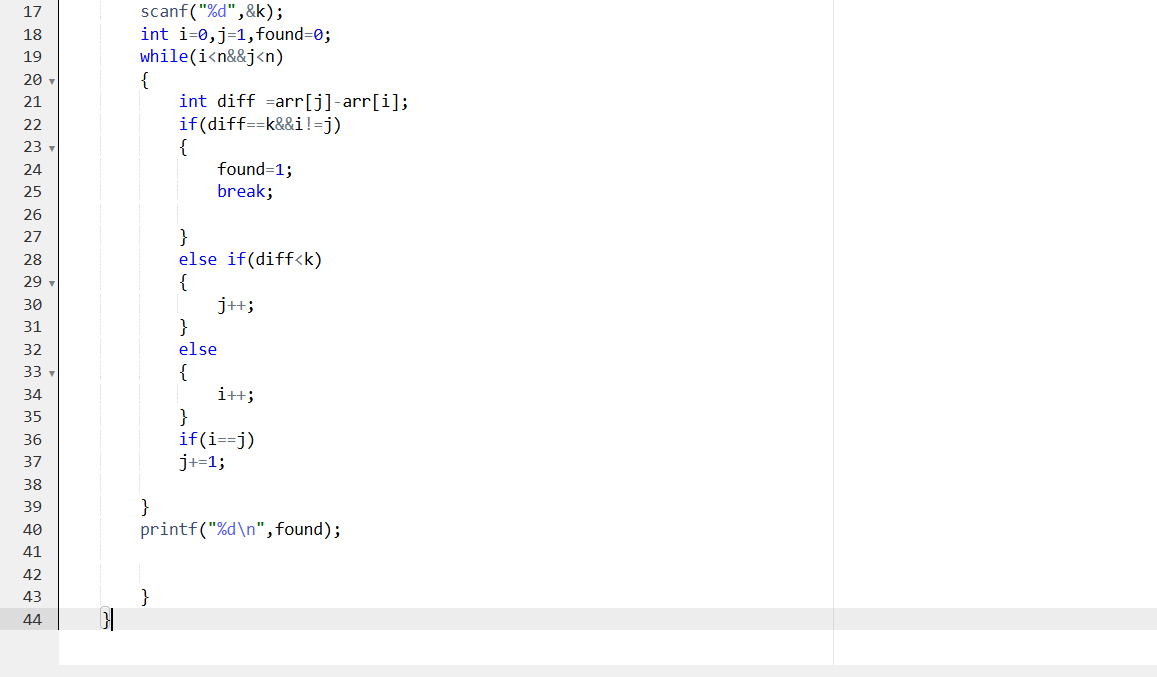
**Week-6**

**241001252**

**Question 1:**

**Given an array A of sorted integers and another non negative integer k, find if there exists 2 indices i and j such that A[i] - A[j] = k, i != j. Input Format 1. First line is number of test cases T. Following T lines contain: 2. N, followed by N integers of the array 3. The non-negative integer k**

**Program:**

**Output:**

****

**Question 2:**

**Sam loves chocolates and starts buying them on the 1st day of the year. Each day of the year, x, is numbered from 1 to Y. On days when x is odd, Sam will buy x chocolates; on days when x is even, Sam will not purchase any chocolates.**

**Complete the code in the editor so that for each day Ni (where 1 ≤ x ≤ N ≤ Y) in array arr, the number of chocolates Sam purchased (during days 1 through N) is printed on a new line. This is a function-only challenge, so input is handled for you by the locked stub code in the editor.**

**Input Format**

**The program takes an array of integers as a parameter.**

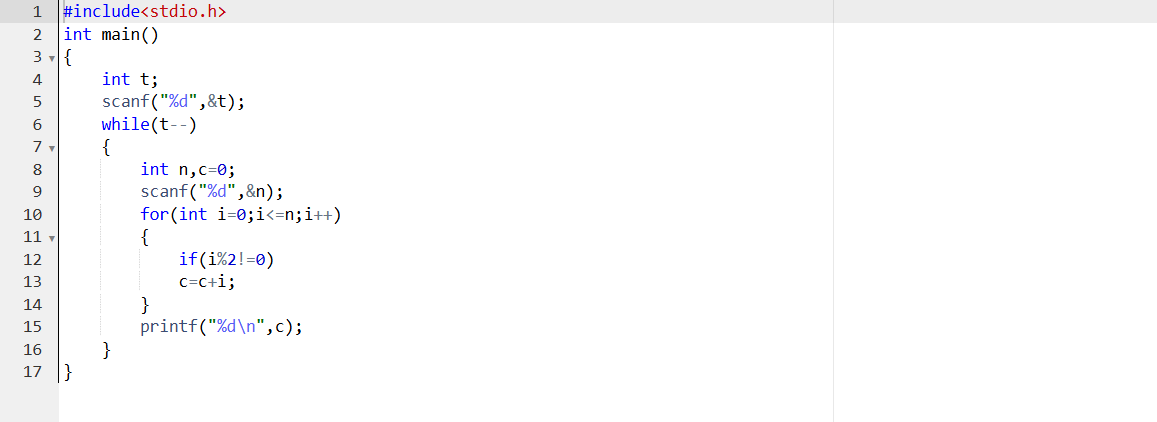
**The locked code in the editor handles reading the following input from stdin, assembling it into an array of integers (arr), and calling calculate(arr).**

**The first line of input contains an integer, T (the number of test cases). Each line i of the T subsequent lines describes the ith test case as an integer, Ni (the number of days).**

**Constraints**

**1 ≤ T ≤ 2 × 105 1 ≤ N ≤ 2 × 106 1 ≤ x ≤ N ≤ Y**

**Program:**

****

**Output:**

****

**Question 3:**

**The number of goals achieved by two football teams in matches in a league is given in the form of two lists. Consider:**

**• Football team A, has played three matches, and has scored { 1 , 2 , 3 } goals in each match respectively.**

**• Football team B, has played two matches, and has scored { 2, 4 } goals in each match respectively. • Your task is to compute, for each match of team B, the total number of matches of team A,**

**• where team A has scored less than or equal to the number of goals scored by team B in that match. In the above case:**

**• For 2 goals scored by team B in its first match, team A has 2 matches with scores 1 and 2.**

**• For 4 goals scored by team B in its second match, team A has 3 matches with scores 1, 2 and 3. Hence, the answer: {2, 3}.**

**Complete the code in the editor below.**

**The program must return an array of m positive integers, one for each maxes[i] representing the total number of elements nums[j] satisfying nums[j] ≤ maxes[i] where 0 ≤ j < n and 0 ≤ i < m, in the given order.**

**It has the following:**

**nums[nums[0],...nums[n-1]]: first array of positive integers**

**maxes[maxes[0],...maxes[n-1]]: second array of positive integers**

**Constraints:**

**2 ≤ n, m ≤ 105, 1 ≤ nums[j] ≤ 109, where 0 ≤ j < n, 1 ≤ maxes[i] ≤ 109, where 0 ≤ i < m.**

**Input Format For Custom Testing**

**Input from stdin will be processed as follows and passed to the function.**

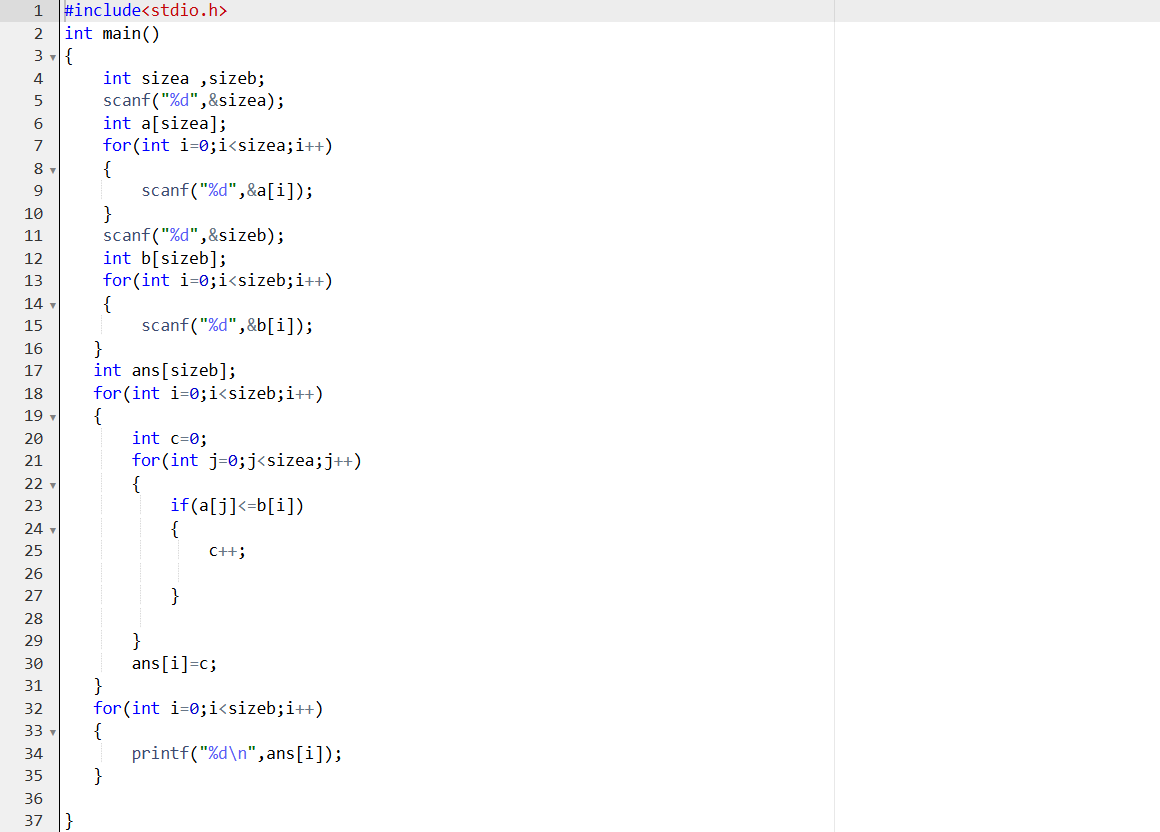
**The first line contains an integer n, the number of elements in nums.**

**The next n lines each contain an integer describing nums[j] where 0 ≤ j < n.**

**The next line contains an integer m, the number of elements in maxes.**

**The next m lines each contain an integer describing maxes[i] where 0 ≤ i < m.**

**Program:**

****

**Output:**

****

**Question 4:**

**A nutritionist is labeling all the best power foods in the market. Every food item arranged in a single line, will have a value beginning from 1 and increasing by 1 for each, until all items have a value associated with them. An item's value is the same as the number of macronutrients it has. For example, food item with value 1 has 1 macronutrient, food item with value 2 has 2 macronutrients, and incrementing in this fashion. The nutritionist has to recommend the best combination to patients, i.e. maximum total of macronutrients. However, the nutritionist must avoid prescribing a particular sum of macronutrients (an 'unhealthy' number), and this sum is known. The nutritionist chooses food items in the increasing order of their value. Compute the highest total of macronutrients that can be prescribed to a patient, without the sum matching the given 'unhealthy' number.**

**Here's an illustration: Given 4 food items (hence value: 1,2,3 and 4), and the unhealthy sum being 6 macronutrients, on choosing items 1, 2, 3 -> the sum is 6, which matches the 'unhealthy' sum. Hence, one of the three needs to be skipped. Thus, the best combination is from among:**

**• 2 + 3 + 4 = 9**

**• 1 + 3 + 4 = 8**

**• 1 + 2 + 4 = 7**

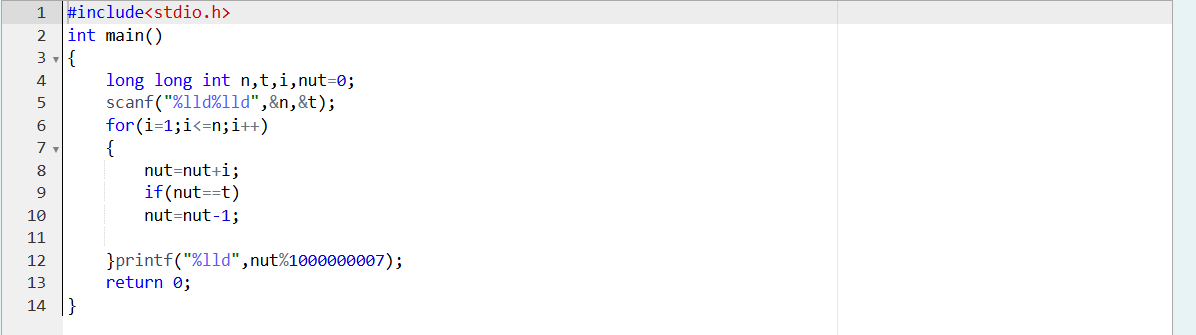
**Since 2 + 3 + 4 = 9, allows for maximum number of macronutrients, 9 is the right answer. Complete the code in the editor below. It must return an integer that represents the maximum total of macronutrients, modulo 1000000007 (109 + 7). It has the following: n: an integer that denotes the number of food items k: an integer that denotes the unhealthy number Constraints**

**• 1 ≤ n ≤ 2 × 109**

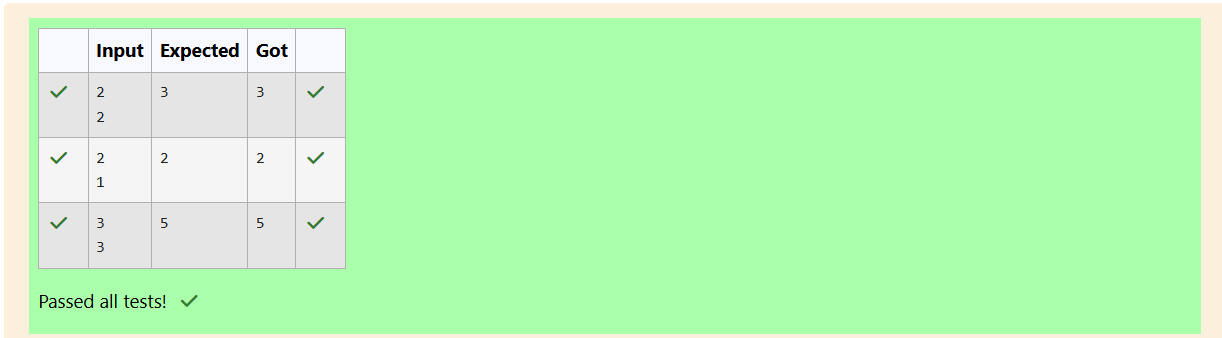
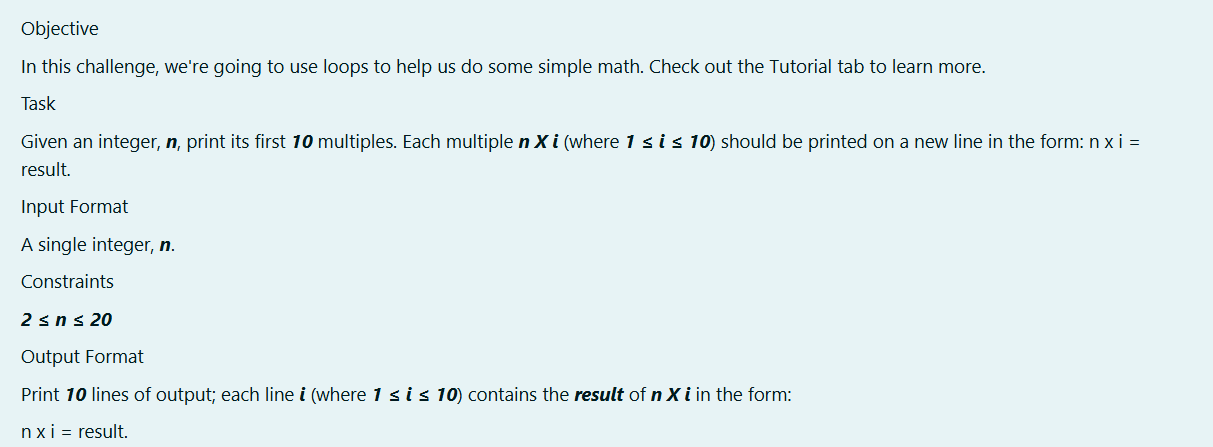
**• 1 ≤ k ≤ 4 × 1015**

**Input Format For Custom Testing**

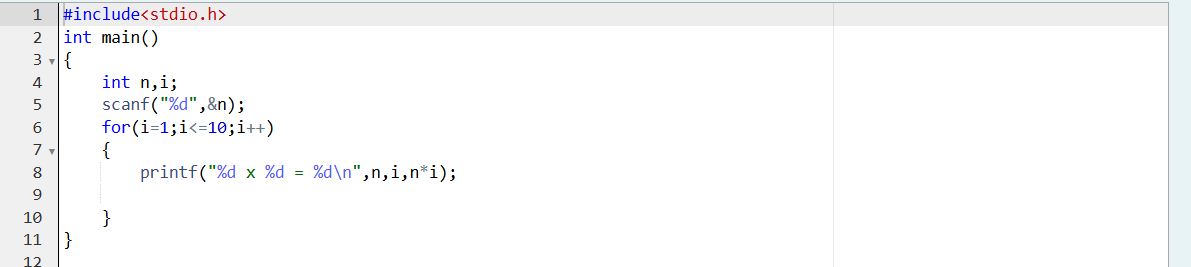
**The first line contains an integer, n, that denotes the number of food items. The second line contains an integer, k, that denotes the unhealthy number.**

**Program:**

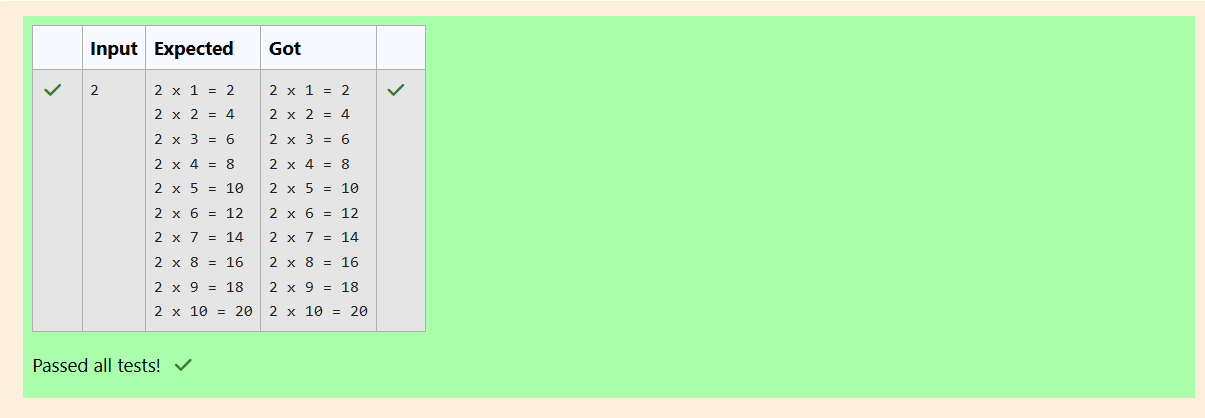
**Output:**

**Question 5: **

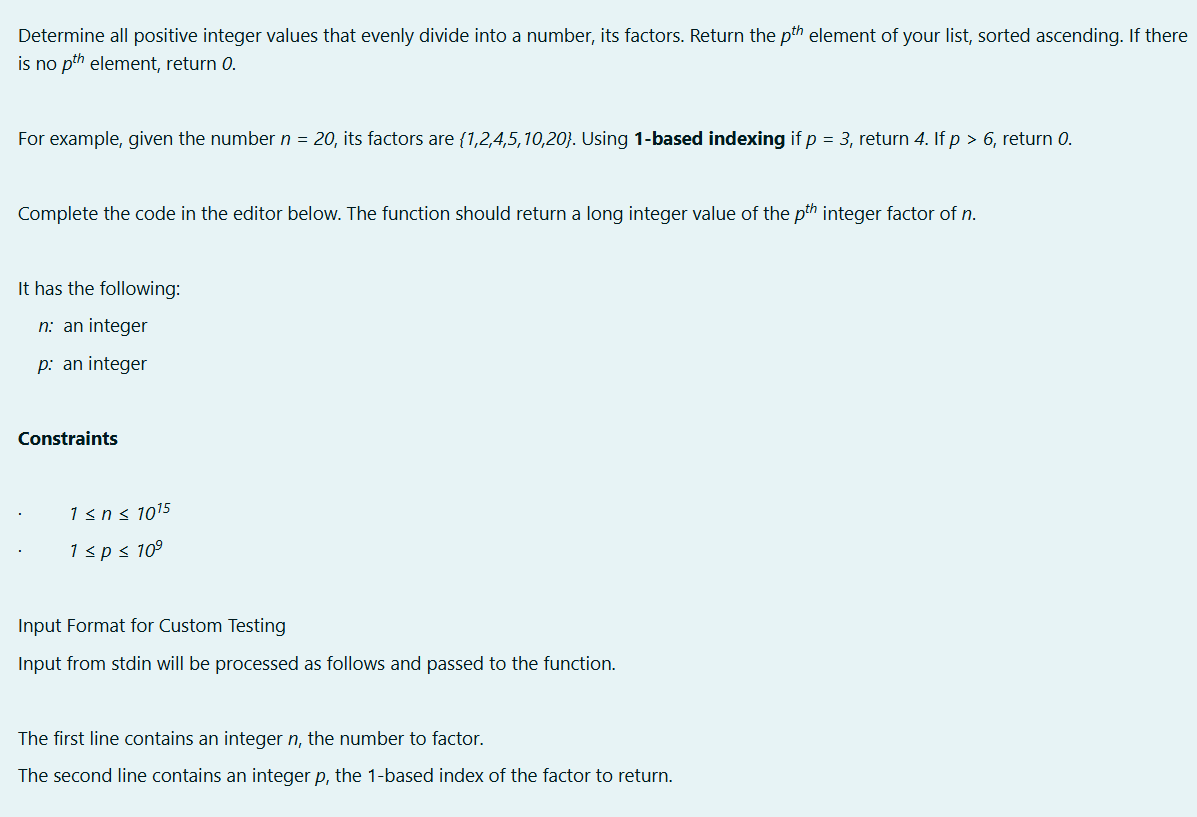
**Program:**

****

**Output:**

****

**Question 6:**

****

**Program&Output:**

****